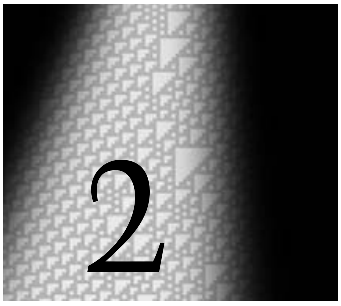


The Crucial Experiment

How Do Simple Programs Behave?

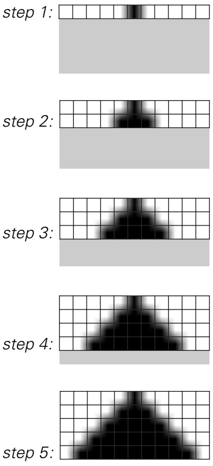
New directions in science have typically been initiated by certain central observations or experiments. And for the kind of science that I describe in this book these concerned the behavior of simple programs.

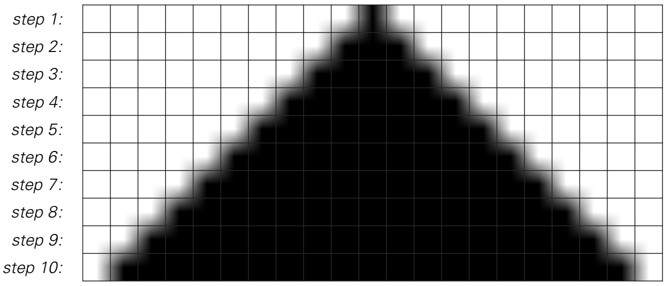
In our everyday experience with computers, the programs that we encounter are normally set up to perform very definite tasks. But the key idea that I had nearly twenty years ago—and that eventually led to the whole new kind of science in this book—was to ask what happens if one instead just looks at simple arbitrarily chosen programs, created without any specific task in mind. How do such programs typically behave?

The mathematical methods that have in the past dominated theoretical science do not help much with such a question. But with a computer it is straightforward to start doing experiments to investigate it. For all one need do is just set up a sequence of possible simple programs, and then run them and see how they behave.

Any program can at some level be thought of as consisting of a set of rules that specify what it should do at each step. There are many possible ways to set up these rules—and indeed we will study quite a few of them in the course of this book. But for now, I will consider a particular class of examples called cellular automata, that were the very first kinds of simple programs that I investigated in the early 1980s.

An important feature of cellular automata is that their behavior can readily be presented in a visual way. And so the picture below shows what one cellular automaton does over the course of ten steps.

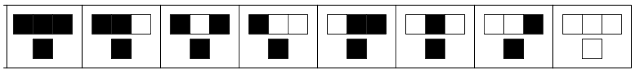




A visual representation of the behavior of a cellular automaton, with each row of cells corresponding to one step. At the first step the cell in the center is black and all other cells are white. Then on each successive step, a particular cell is made black whenever it or either of its neighbors were black on the step before. As the picture shows, this leads to a simple expanding pattern uniformly filled with black.

The cellular automaton consists of a line of cells, each colored either black or white. At every step there is then a definite rule that determines the color of a given cell from the color of that cell and its immediate left and right neighbors on the step before.

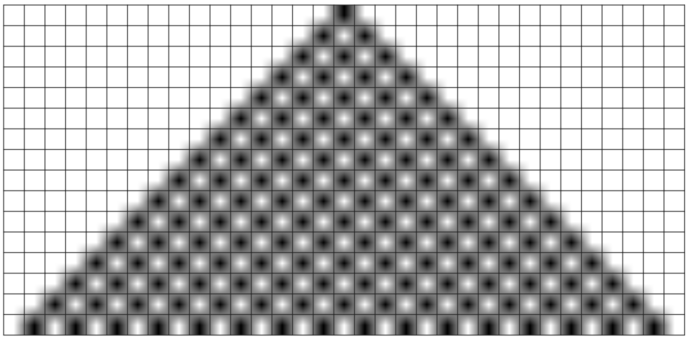
For the particular cellular automaton shown here the rule specifies—as in the picture below—that a cell should be black in all cases where it or either of its neighbors were black on the step before.

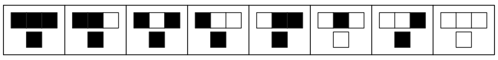


A representation of the rule for the cellular automaton shown above. The top row in each box gives one of the possible combinations of colors for a cell and its immediate neighbors. The bottom row then specifies what color the center cell should be on the next step in each of these cases. In the numbering scheme described in Chapter 3, this is cellular automaton rule 254.

And the picture at the top of the page shows that starting with a single black cell in the center this rule then leads to a simple growing pattern uniformly filled with black. But modifying the rule just slightly one can immediately get a different pattern.

As a first example, the picture at the top of the facing page shows what happens with a rule that makes a cell white whenever both of its neighbors were white on the step before—even if the cell itself was black before. And rather than producing a pattern that is uniformly filled with black, this rule now instead gives a pattern that repeatedly alternates between black and white like a checkerboard.

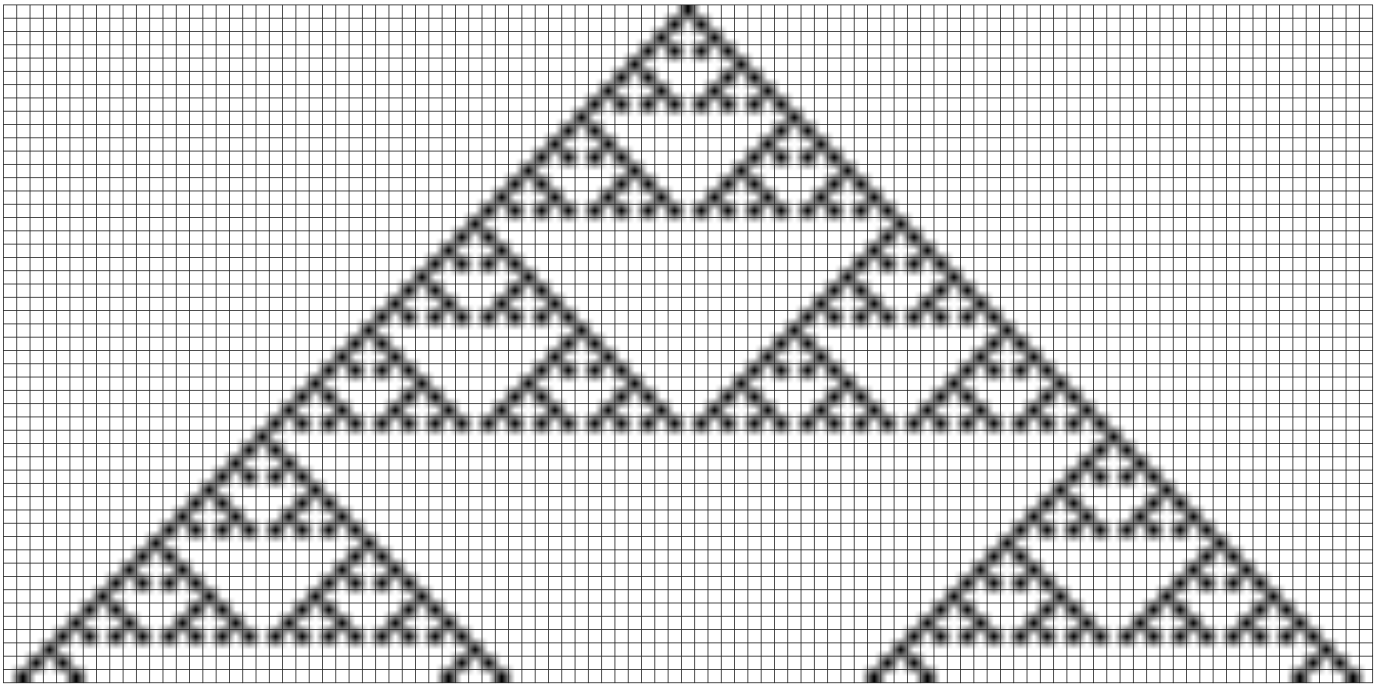


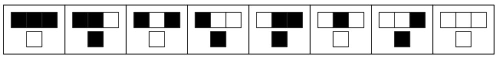


A cellular automaton with a slightly different rule. The rule makes a particular cell black if either of its neighbors was black on the step before, and makes the cell white if both its neighbors were white. Starting from a single black cell, this rule leads to a checkerboard pattern. In the numbering scheme of Chapter 3, this is cellular automaton rule 250.

This pattern is however again fairly simple. And we might assume that at least with the type of cellular automata that we are considering, any rule we might choose would always give a pattern that is quite simple. But now we are in for our first surprise.

The picture below shows the pattern produced by a cellular automaton of the same type as before, but with a slightly different rule.

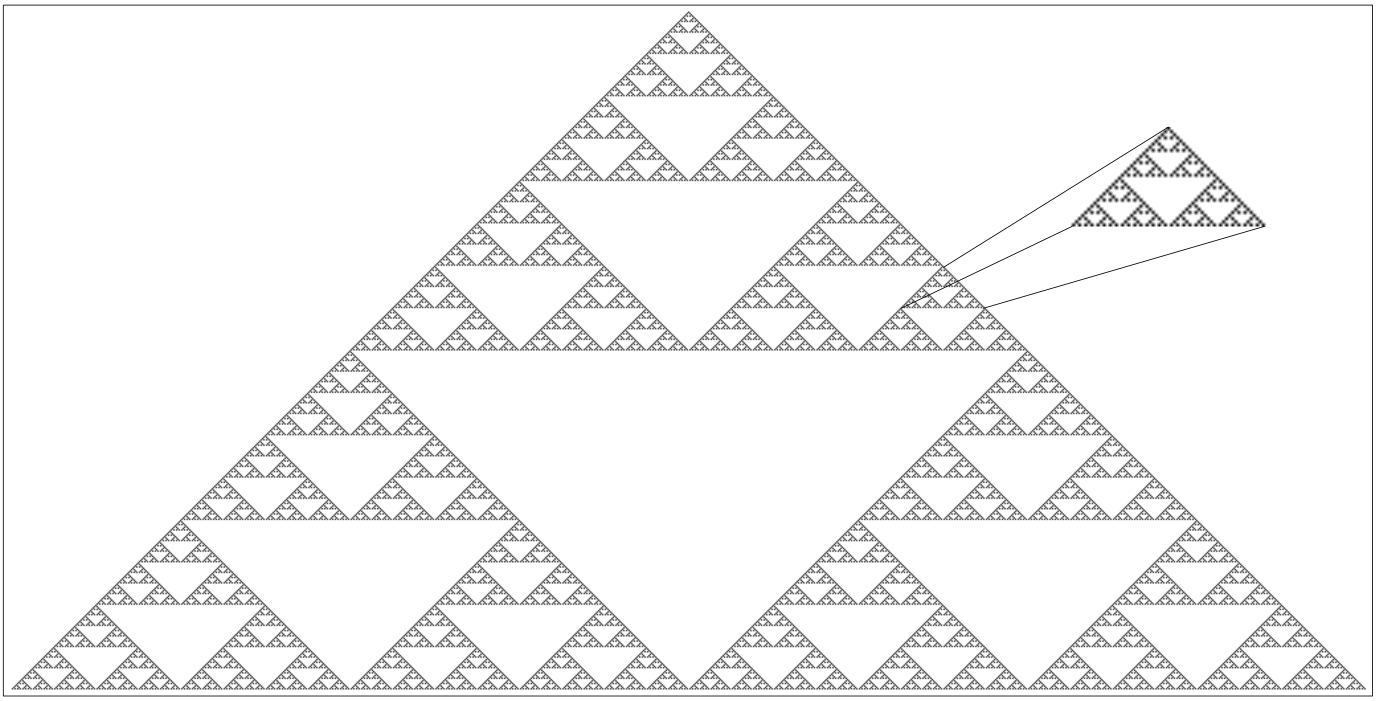




A cellular automaton that produces an intricate nested pattern. The rule in this case is that a cell should be black whenever one or the other, but not both, of its neighbors were black on the step before. Even though the rule is very simple, the picture shows that the overall pattern obtained over the course of 50 steps starting from a single black cell is not so simple. The particular rule used here can be described by the formula $a_i' = Mod[a_{i-1} + a_{i+1}, 2]$. In the numbering scheme of Chapter 3, it is cellular automaton rule 90.

This time the rule specifies that a cell should be black when either its left neighbor or its right neighbor—but not both—were black on the step before. And again this rule is undeniably quite simple. But now the picture shows that the pattern it produces is not so simple.

And if one runs the cellular automaton for more steps, as in the picture below, then a rather intricate pattern emerges. But one can now see that this pattern has very definite regularity. For even though it is intricate, one can see that it actually consists of many nested triangular pieces that all have exactly the same form. And as the picture shows, each of these pieces is essentially just a smaller copy of the whole pattern, with still smaller copies nested in a very regular way inside it.

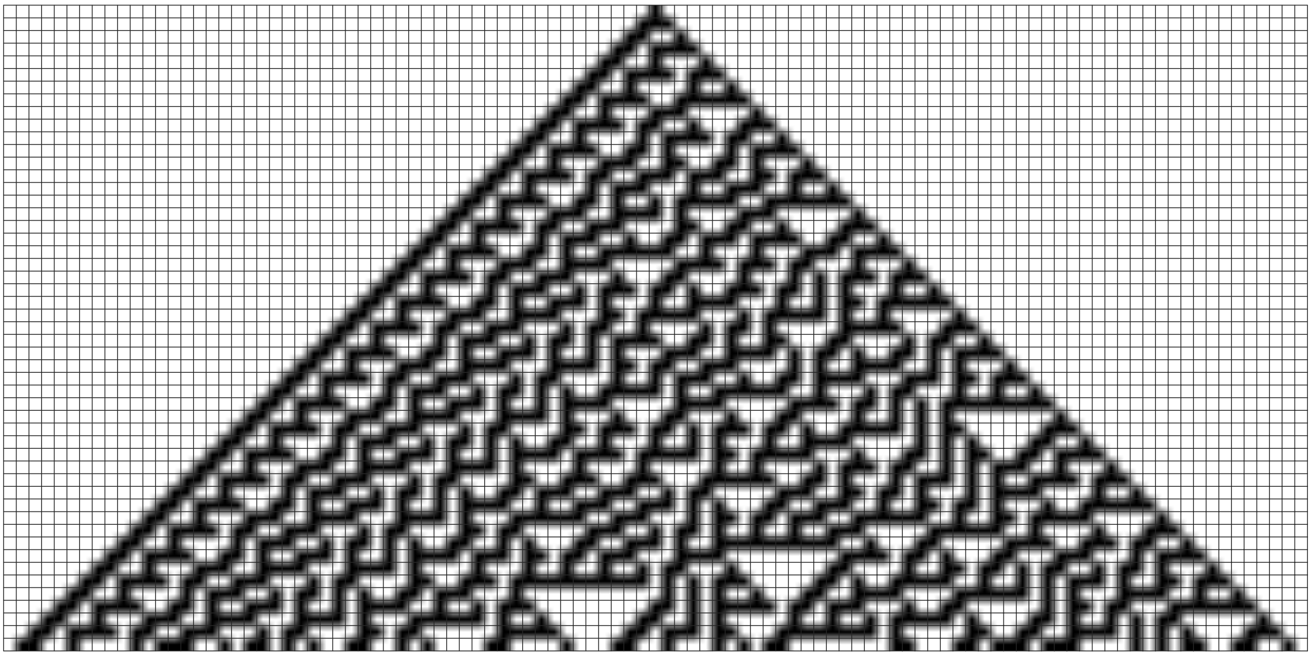


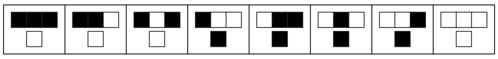
A larger version of the pattern from the previous page, now shown without a grid explicitly indicating each cell. The picture shows five hundred steps of cellular automaton evolution. The pattern obtained is intricate, but has a definite nested structure. Indeed, as the picture illustrates, each triangular section is essentially just a smaller copy of the whole pattern, with still smaller copies nested inside it. Patterns with nested structure of this kind are often called "fractal" or "self-similar".

So of the three cellular automata that we have seen so far, all ultimately yield patterns that are highly regular: the first a simple uniform pattern, the second a repetitive pattern, and the third an intricate but still nested pattern. And we might assume that at least for cellular automata with rules as simple as the ones we have been using these three forms of behavior would be all that we could ever get.

But the remarkable fact is that this turns out to be wrong.

And the picture below shows an example of this. The rule used—that I call rule 30—is of exactly the same kind as before, and can be described as follows. First, look at each cell and its right-hand neighbor. If both of these were white on the previous step, then take the new color of the cell to be whatever the previous color of its left-hand neighbor was. Otherwise, take the new color to be the opposite of that.





A cellular automaton with a simple rule that generates a pattern which seems in many respects random. The rule used is of the same type as in the previous examples, and the cellular automaton is again started from a single black cell. But now the pattern that is obtained is highly complex, and shows almost no overall regularity. This picture is our first example of the fundamental phenomenon that even with simple underlying rules and simple initial conditions, it is possible to produce behavior of great complexity. In the numbering scheme of Chapter 3, the cellular automaton shown here is rule 30.

The picture shows what happens when one starts with just one black cell and then applies this rule over and over again. And what one sees is something quite startling—and probably the single most surprising scientific discovery I have ever made. Rather than getting a simple regular pattern as we might expect, the cellular automaton instead produces a pattern that seems extremely irregular and complex.

But where does this complexity come from? We certainly did not put it into the system in any direct way when we set it up. For we just used a simple cellular automaton rule, and just started from a simple initial condition containing a single black cell.

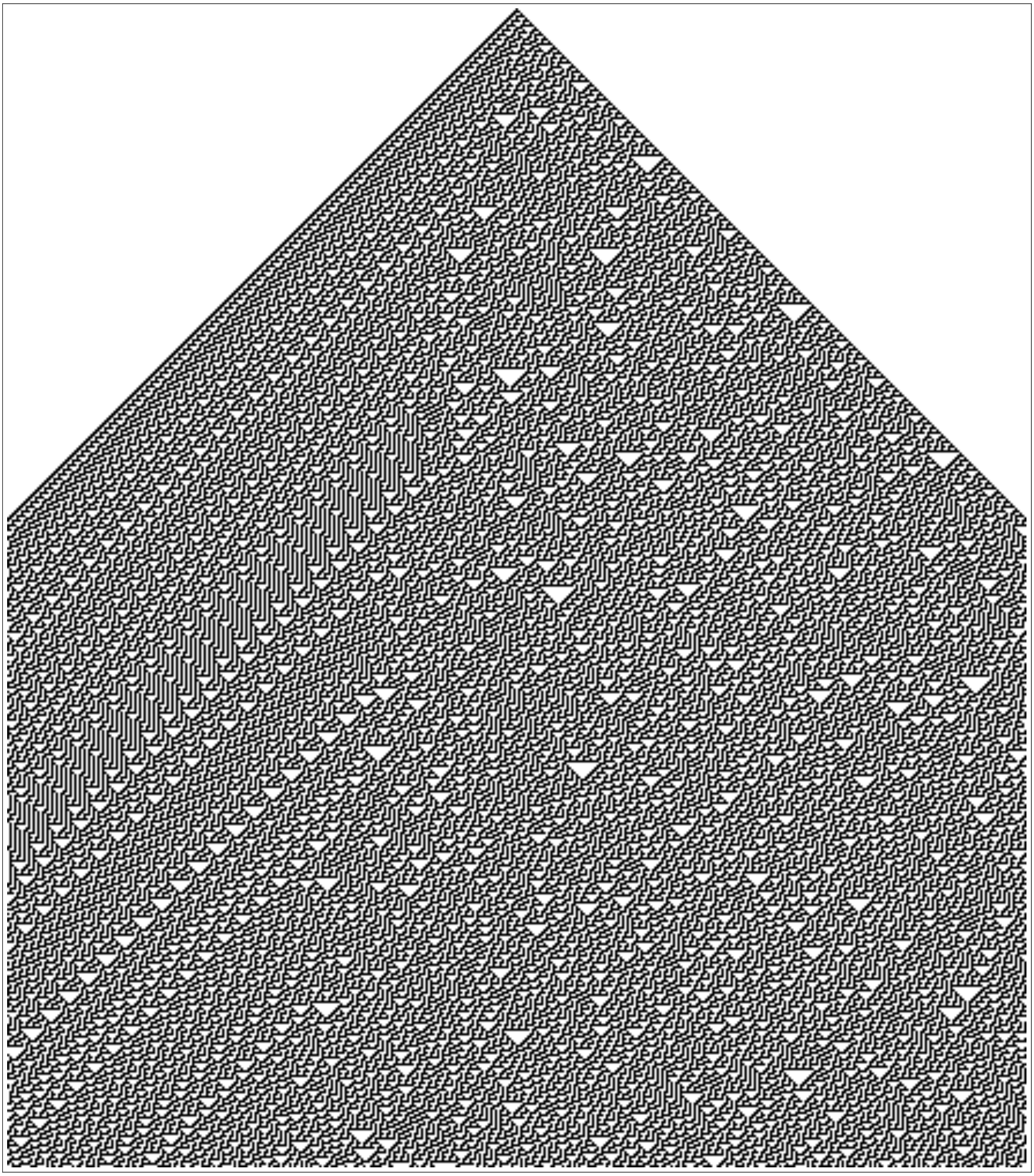
Yet the picture shows that despite this, there is great complexity in the behavior that emerges. And indeed what we have seen here is a first example of an extremely general and fundamental phenomenon that is at the very core of the new kind of science that I develop in this book. Over and over again we will see the same kind of thing: that even though the underlying rules for a system are simple, and even though the system is started from simple initial conditions, the behavior that the system shows can nevertheless be highly complex. And I will argue that it is this basic phenomenon that is ultimately responsible for most of the complexity that we see in nature.

The next two pages show progressively more steps in the evolution of the rule 30 cellular automaton from the previous page. One might have thought that after maybe a thousand steps the behavior would eventually resolve into something simple. But the pictures on the next two pages show that nothing of the sort happens.

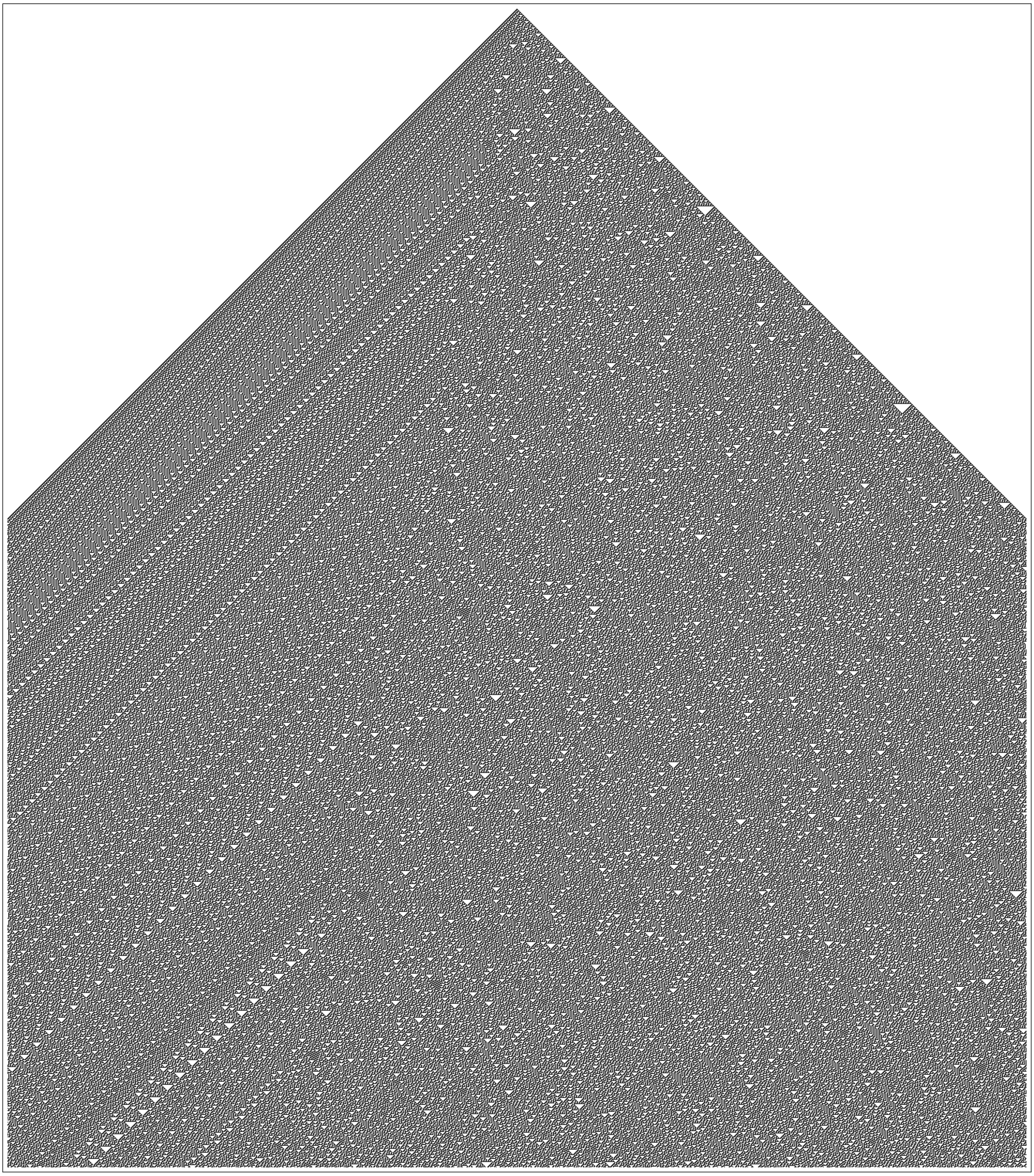
Some regularities can nevertheless be seen. On the left-hand side, for example, there are obvious diagonal bands. And dotted throughout there are various white triangles and other small structures. Yet given the simplicity of the underlying rule, one would expect vastly more regularities. And perhaps one might imagine that our failure to see any in the pictures on the next two pages is just a reflection of some kind of inadequacy in the human visual system.

But it turns out that even the most sophisticated mathematical and statistical methods of analysis seem to do no better. For example, one can look at the sequence of colors directly below the initial black cell. And in the first million steps in this sequence, for example, it never repeats, and indeed none of the tests I have ever done on it show any meaningful deviation at all from perfect randomness.

In a sense, however, there is a certain simplicity to such perfect randomness. For even though it may be impossible to predict what



Five hundred steps in the evolution of the rule 30 cellular automaton from page 27. The pattern produced continues to expand on both left and right, but only the part that fits across the page is shown here. The asymmetry between the left and right-hand sides is a direct consequence of asymmetry that exists in the particular underlying cellular automaton rule used.



Fifteen hundred steps of rule 30 evolution. Some regularities are evident, particularly on the left. But even after all these steps there are no signs of overall regularity—and indeed even continuing for a million steps many aspects of the pattern obtained seem perfectly random according to standard mathematical and statistical tests. The picture here shows a total of just under two million individual cells.

color will occur at any specific step, one still knows for example that black and white will on average always occur equally often.

But it turns out that there are cellular automata whose behavior is in effect still more complex—and in which even such averages become very difficult to predict. The pictures on the next several pages give a rather dramatic example. The basic form of the rule is just the same as before. But now the specific rule used—that I call rule 110—takes the new color of a cell to be black in every case except when the previous colors of the cell and its two neighbors were all the same, or when the left neighbor was black and the cell and its right neighbor were both white.

The pattern obtained with this rule shows a remarkable mixture of regularity and irregularity. More or less throughout, there is a very regular background texture that consists of an array of small white triangles repeating every 7 steps. And beginning near the left-hand edge, there are diagonal stripes that occur at intervals of exactly 80 steps.

But on the right-hand side, the pattern is much less regular. Indeed, for the first few hundred steps there is a region that seems essentially random. But by the bottom of the first page, all that remains of this region is three copies of a rather simple repetitive structure.

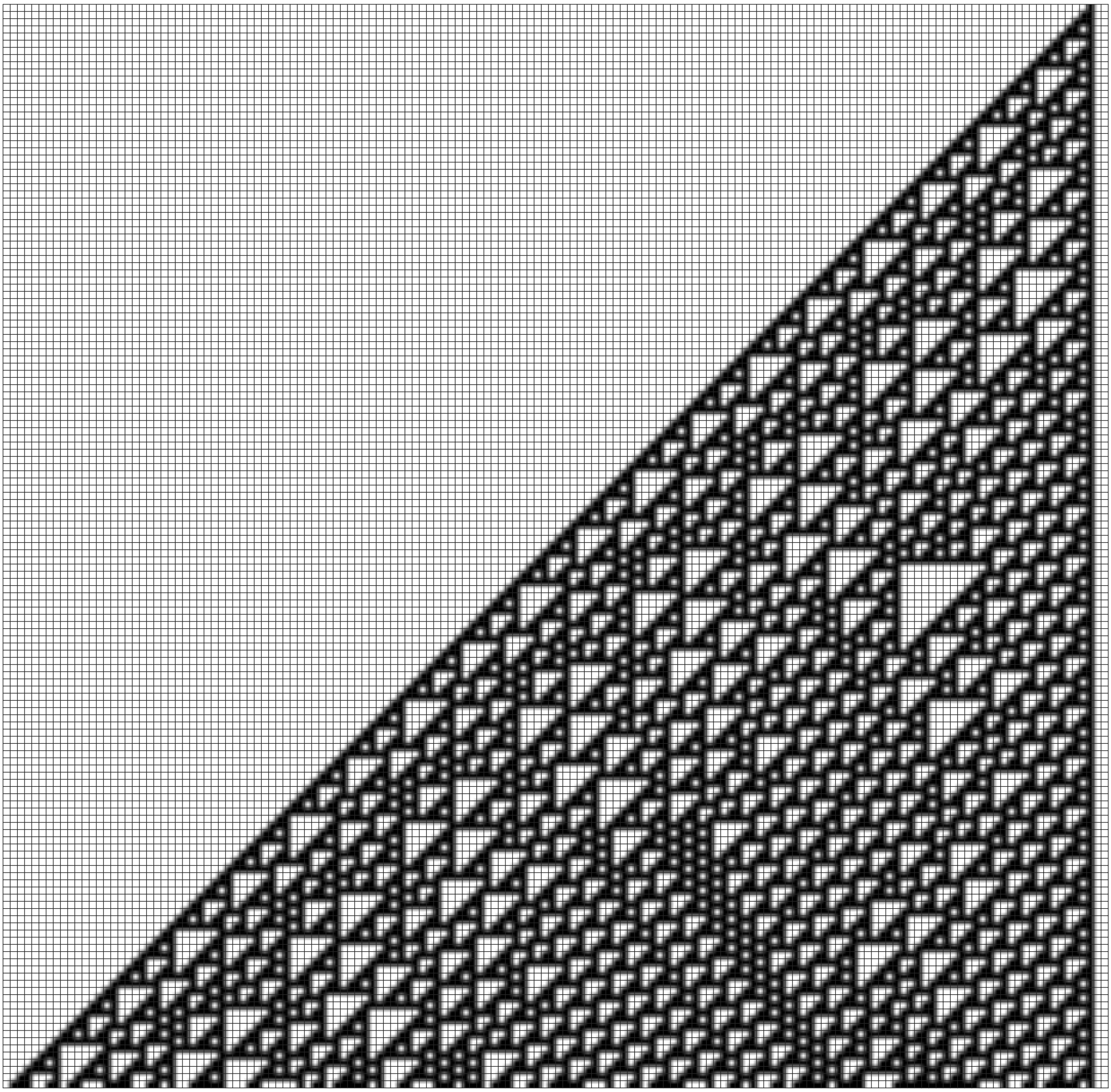
Yet at the top of the next page the arrival of a diagonal stripe from the left sets off more complicated behavior again. And as the system progresses, a variety of definite localized structures are produced.

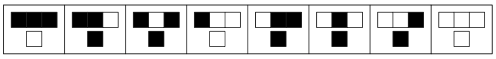
Some of these structures remain stationary, like those at the bottom of the first page, while others move steadily to the right or left at various speeds. And on their own, each of these structures works in a fairly simple way. But as the pictures illustrate, their various interactions can have very complicated effects.

And as a result it becomes almost impossible to predict—even approximately—what the cellular automaton will do.

Will all the structures that are produced eventually annihilate each other, leaving only a very regular pattern? Or will more and more structures appear until the whole pattern becomes quite random?

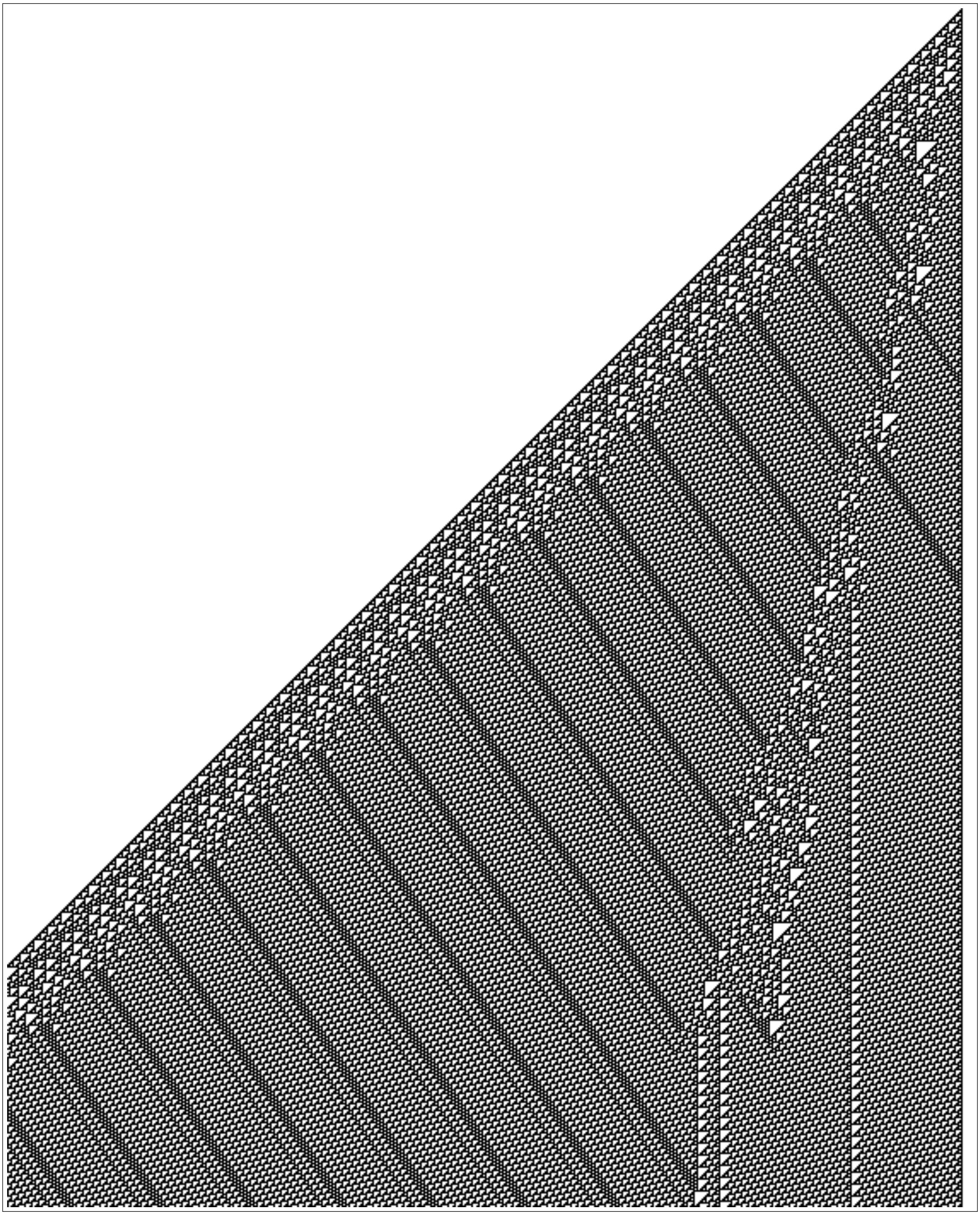
The only sure way to answer these questions, it seems, is just to run the cellular automaton for as many steps as are needed, and to

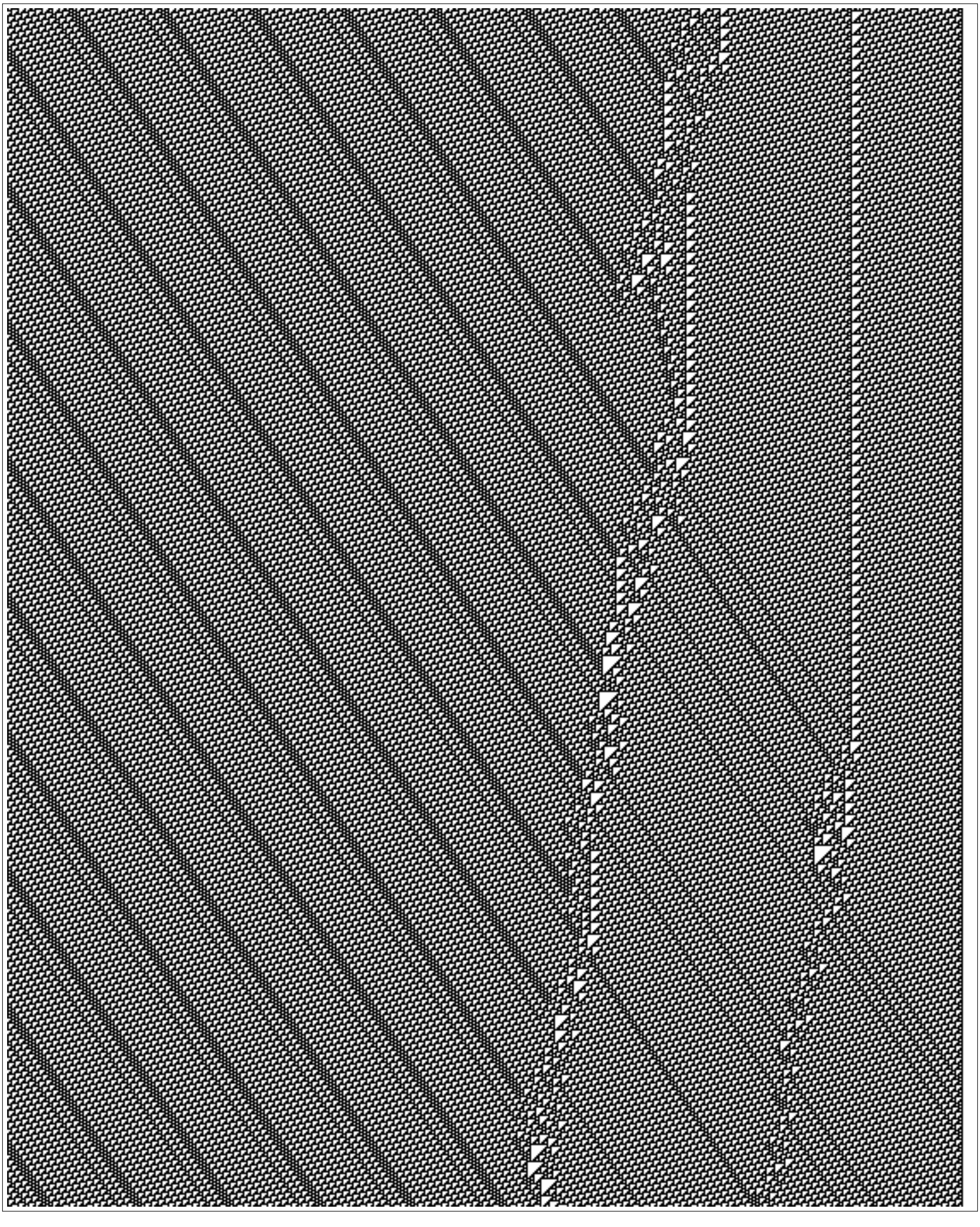


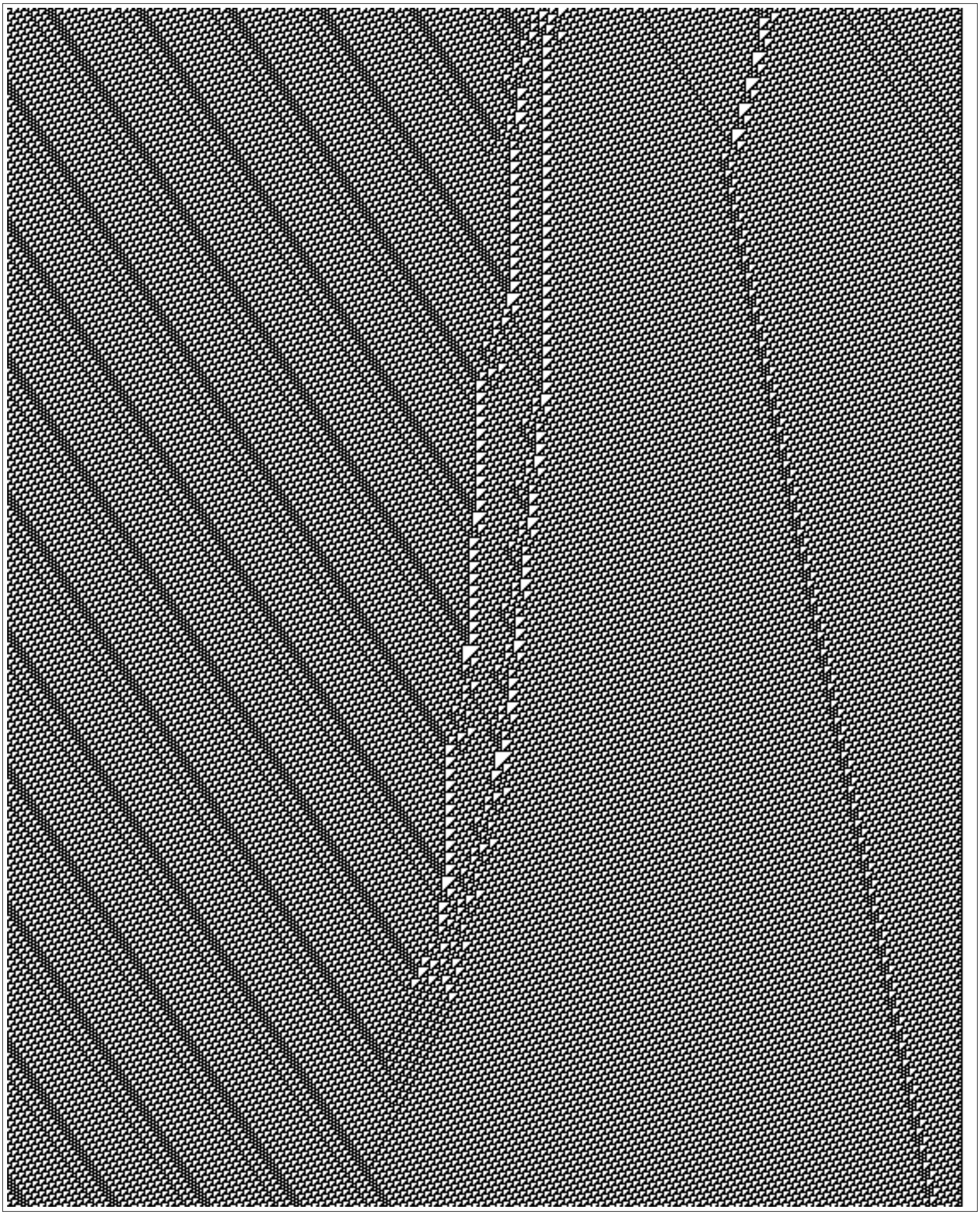


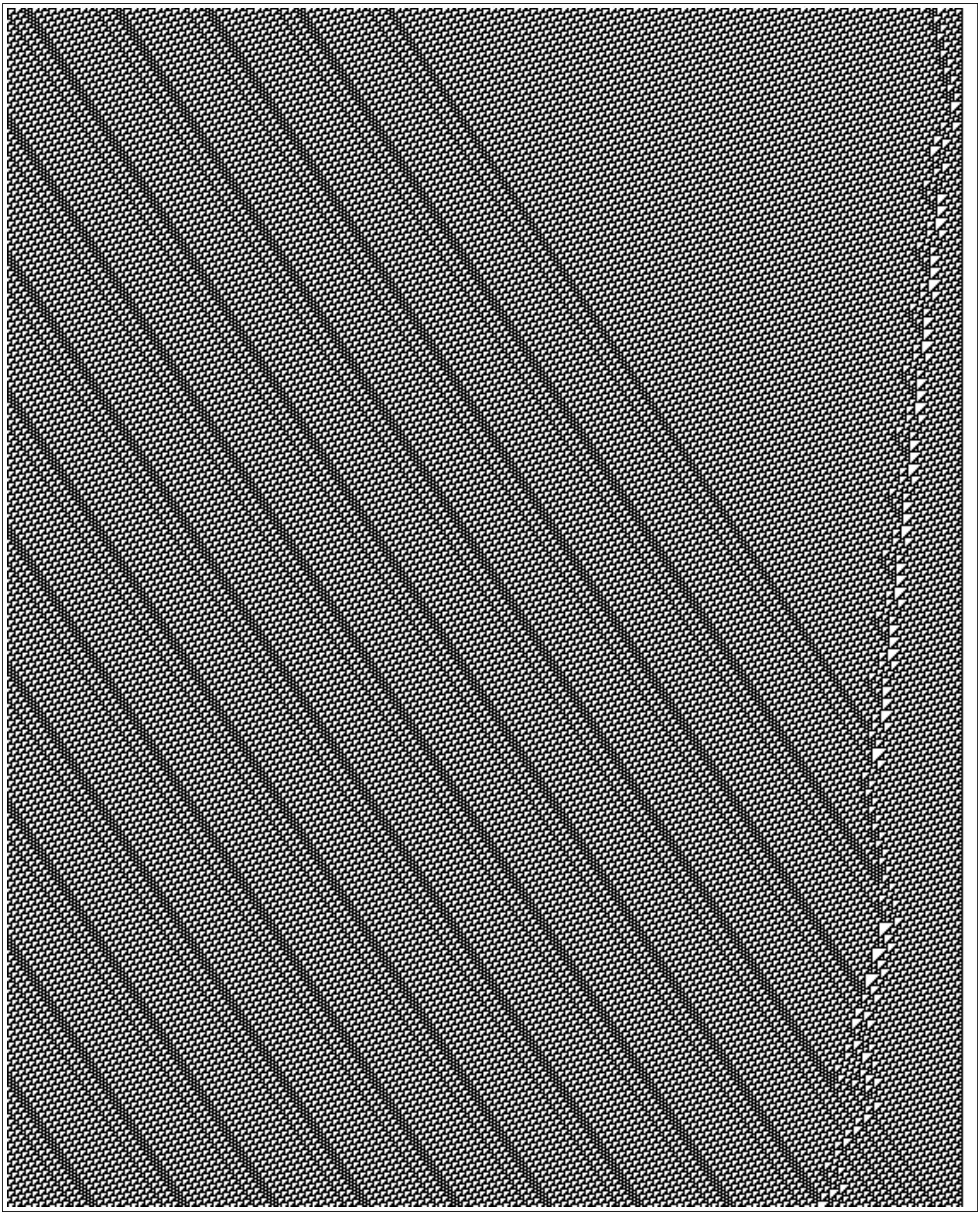
A cellular automaton whose behavior seems neither highly regular nor completely random. The picture is obtained by applying the simple rule shown for a total of 150 steps, starting with a single black cell. Note that the particular rule used here yields a pattern that expands on the left but not on the right. In the scheme defined in Chapter 3, the rule is number 110.

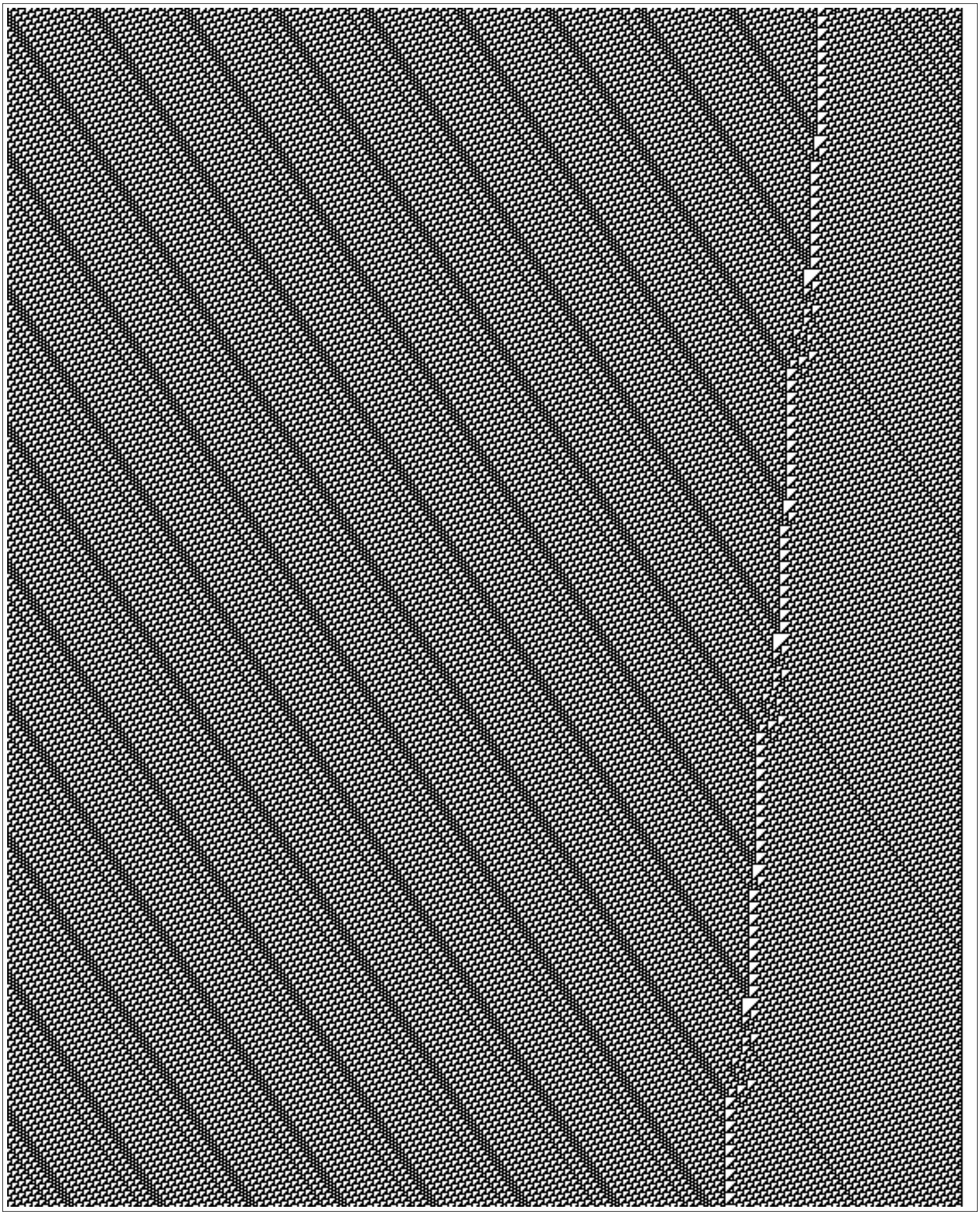
More steps in the pattern shown above. Each successive page shows a total of 700 steps. The pattern continues to expand on the left forever, but only the part that fits across each page is shown. For a long time it is not clear how the right-hand part of the pattern will eventually look. But after 2780 steps, a fairly simple repetitive structure emerges. Note that to generate the pictures that follow requires applying the underlying cellular automaton rule for individual cells a total of about 12 million times. ▶

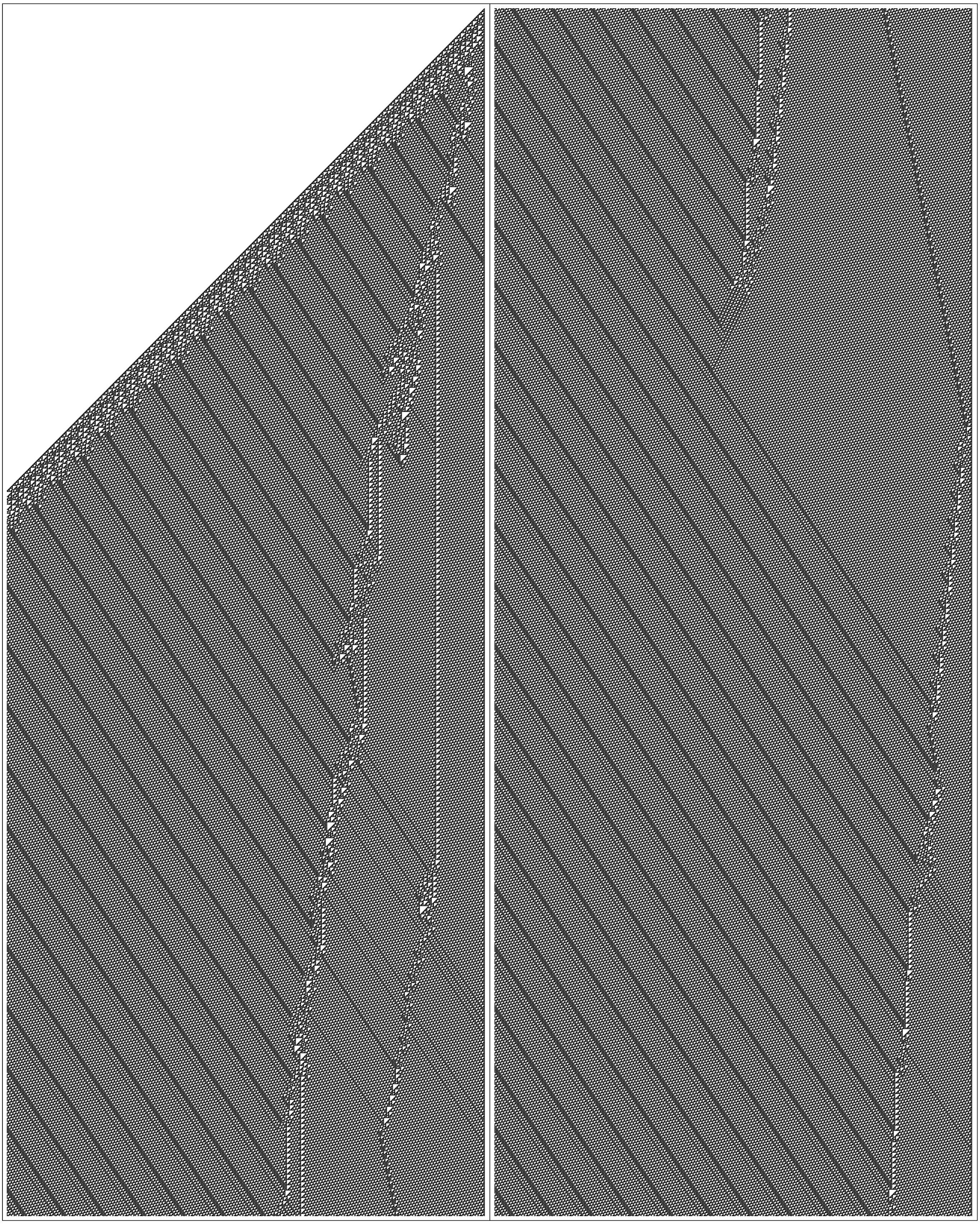












watch what happens. And as it turns out, in the particular case shown here, the outcome is finally clear after about 2780 steps: one structure survives, and that structure interacts with the periodic stripes coming from the left to produce behavior that repeats every 240 steps.

However certain one might be that simple programs could never do more than produce simple behavior, the pictures on the past few pages should forever disabuse one of that notion. And indeed, what is perhaps most bizarre about the pictures is just how little trace they ultimately show of the simplicity of the underlying cellular automaton rule that was used to produce them.

One might think, for example, that the fact that all the cells in a cellular automaton follow exactly the same rule would mean that in pictures like the last few pages all cells would somehow obviously be doing the same thing. But instead, they seem to be doing quite different things. Some of them, for example, are part of the regular background, while others are part of one or another localized structure. And what makes this possible is that even though individual cells follow the same rule, different configurations of cells with different sequences of colors can together produce all sorts of different kinds of behavior.

Looking just at the original cellular automaton rule one would have no realistic way to foresee all of this. But by doing the appropriate computer experiments one can easily find out what actually happens—and in effect begin the process of exploring a whole new world of remarkable phenomena associated with simple programs.

The Need for a New Intuition

The pictures in the previous section plainly show that it takes only very simple rules to produce highly complex behavior. Yet at first this may seem almost impossible to believe. For it goes against some of our most basic intuition about the way things normally work.

◀ A single picture of the behavior from the previous five pages. A total of 3200 steps are shown. Note that this is more than twice as many as in the picture on page 30.

For our everyday experience has led us to expect that an object that looks complicated must have been constructed in a complicated way. And so, for example, if we see a complicated mechanical device, we normally assume that the plans from which the device was built must also somehow be correspondingly complicated.

But the results at the end of the previous section show that at least sometimes such an assumption can be completely wrong. For the patterns we saw are in effect built according to very simple plans—that just tell us to start with a single black cell, and then repeatedly to apply a simple cellular automaton rule. Yet what emerges from these plans shows an immense level of complexity.

So what is it that makes our normal intuition fail? The most important point seems to be that it is mostly derived from experience with building things and doing engineering—where it so happens that one avoids encountering systems like the ones in the previous section.

For normally we start from whatever behavior we want to get, then try to design a system that will produce it. Yet to do this reliably, we have to restrict ourselves to systems whose behavior we can readily understand and predict—for unless we can foresee how a system will behave, we cannot be sure that the system will do what we want.

But unlike engineering, nature operates under no such constraint. So there is nothing to stop systems like those at the end of the previous section from showing up. And in fact one of the important conclusions of this book is that such systems are actually very common in nature.

But because the only situations in which we are routinely aware both of underlying rules and overall behavior are ones in which we are building things or doing engineering, we never normally get any intuition about systems like the ones at the end of the previous section.

So is there then any aspect of everyday experience that should give us a hint about the phenomena that occur in these systems? Probably the closest is thinking about features of practical computing.

For we know that computers can perform many complex tasks. Yet at the level of basic hardware a typical computer is capable of executing just a few tens of kinds of simple logical, arithmetic and other instructions. And to some extent the fact that by executing large numbers of such instructions one can get all sorts of complex behavior is similar to the phenomenon we have seen in cellular automata.

But there is an important difference. For while the individual machine instructions executed by a computer may be quite simple, the sequence of such instructions defined by a program may be long and complicated. And indeed—much as in other areas of engineering—the typical experience in developing software is that to make a computer do something complicated requires setting up a program that is itself somehow correspondingly complicated.

In a system like a cellular automaton the underlying rules can be thought of as rough analogs of the machine instructions for a computer, while the initial conditions can be thought of as rough analogs of the program. Yet what we saw in the previous section is that in cellular automata not only can the underlying rules be simple, but the initial conditions can also be simple—consisting say of just a single black cell—and still the behavior that is produced can be highly complex.

So while practical computing gives a hint of part of what we saw in the previous section, the whole phenomenon is something much larger and stronger. And in a sense the most puzzling aspect of it is that it seems to involve getting something from nothing.

For the cellular automata we set up are by any measure simple to describe. Yet when we ran them we ended with patterns so complex that they seemed to defy any simple description at all.

And one might hope that it would be possible to call on some existing kind of intuition to understand such a fundamental phenomenon. But in fact there seems to be no branch of everyday experience that provides what is needed. And so we have no choice but to try to develop a whole new kind of intuition.

And the only reasonable way to do this is to expose ourselves to a large number of examples. We have seen so far only a few examples, all in cellular automata. But in the next few chapters we will see many more examples, both in cellular automata and in all sorts of other systems. And by absorbing these examples, one is in the end able to develop an intuition that makes the basic phenomena that I have discovered seem somehow almost obvious and inevitable.

Why These Discoveries Were Not Made Before

The main result of this chapter—that programs based on simple rules can produce behavior of great complexity—seems so fundamental that one might assume it must have been discovered long ago. But it was not, and it is useful to understand some of the reasons why it was not.

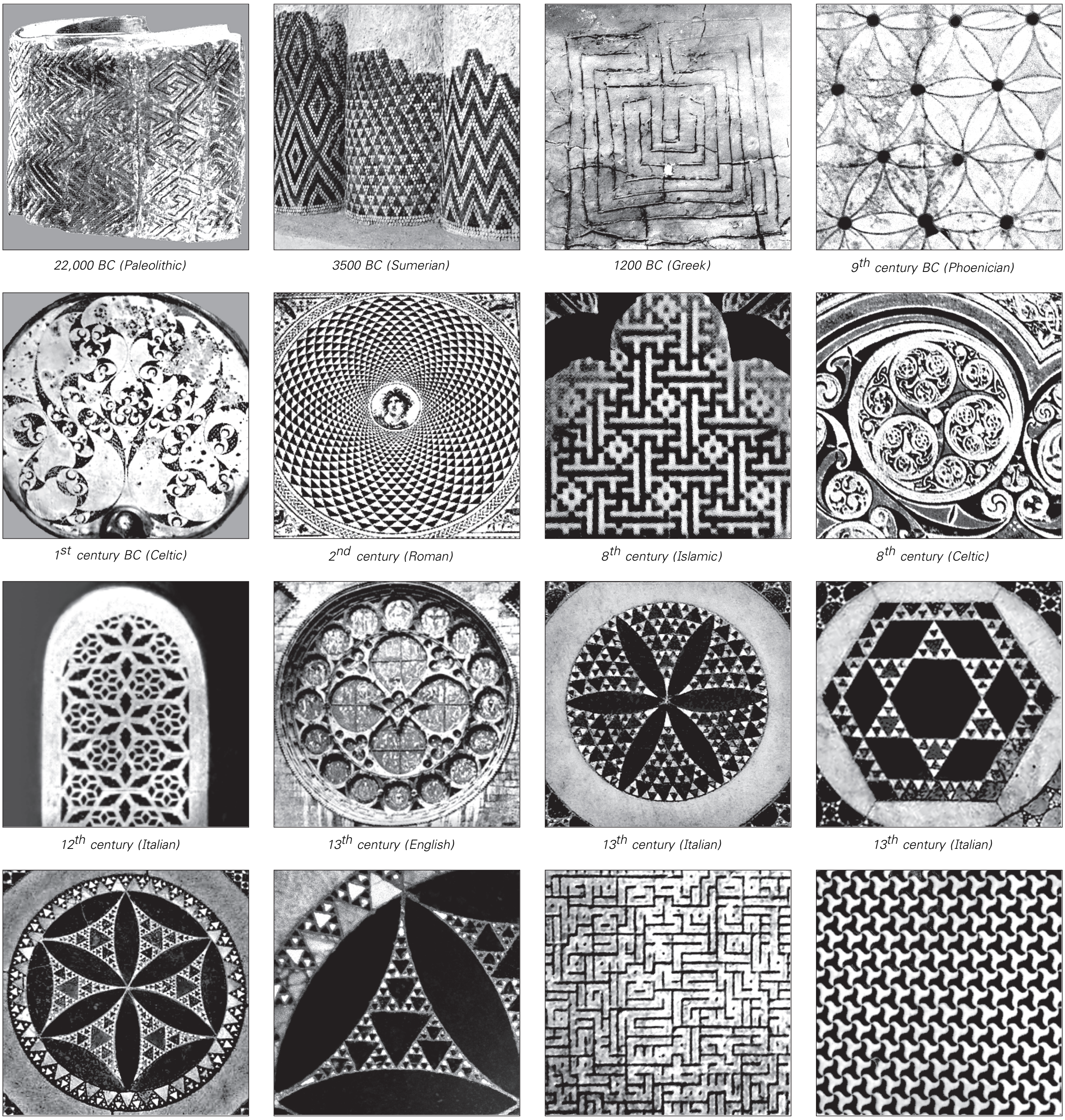
In the history of science it is fairly common that new technologies are ultimately what make new areas of basic science develop. And thus, for example, telescope technology was what led to modern astronomy, and microscope technology to modern biology. And now, in much the same way, it is computer technology that has led to the new kind of science that I describe in this book.

Indeed, this chapter and several of those that follow can in a sense be viewed as an account of some of the very simplest experiments that can be done using computers. But why is it that such simple experiments were never done before?

One reason is just that they were not in the mainstream of any existing field of science or mathematics. But a more important reason is that standard intuition in traditional science gave no reason to think that their results would be interesting.

And indeed, if it had been known that they were worthwhile, many of the experiments could actually have been done even long before computers existed. For while it may be somewhat tedious, it is certainly possible to work out the behavior of something like a cellular automaton by hand. And in fact, to do so requires absolutely no sophisticated ideas from mathematics or elsewhere: all it takes is an understanding of how to apply simple rules repeatedly.

And looking at the historical examples of ornamental art on the facing page, there seems little reason to think that the behavior of many cellular automata could not have been worked out many centuries or even millennia ago. And perhaps one day some Babylonian artifact created using the rule 30 cellular automaton from page 27 will be unearthed. But I very much doubt it. For I tend to think that if pictures like the one on page 27 had ever in fact been seen in ancient times then science would have been led down a very different path from the one it actually took.



22,000 BC (Paleolithic)

3500 BC (Sumerian)

1200 BC (Greek)

9th century BC (Phoenician)

1st century BC (Celtic)

2nd century (Roman)

8th century (Islamic)

8th century (Celtic)

12th century (Italian)

13th century (English)

13th century (Italian)

13th century (Italian)

13th century (Italian)

13th century (Italian)

14th century (Islamic)

14th century (Islamic)

Historical examples of ornamental art. Repetitive patterns are common and some nested patterns are seen, but the more complicated kinds of patterns discussed in this chapter do not ever appear to have been used. Note that the second-to-last picture is not an abstract design, but is instead text written in a highly stylized form of Arabic script.

Even early in antiquity attempts were presumably made to see whether simple abstract rules could reproduce the behavior of natural systems. But so far as one can tell the only types of rules that were tried were ones associated with standard geometry and arithmetic. And using these kinds of rules, only rather simple behavior could be obtained—adequate to explain some of the regularities observed in astronomy, but unable to capture much of what is seen elsewhere in nature.

And perhaps because of this, it typically came to be assumed that a great many aspects of the natural world are simply beyond human understanding. But finally the successes based on calculus in the late 1600s began to overthrow this belief. For with calculus there was finally real success in taking abstract rules created by human thought and using them to reproduce all sorts of phenomena in the natural world.

But the particular rules that were found to work were fairly sophisticated ones based on particular kinds of mathematical equations. And from seeing the sophistication of these rules there began to develop an implicit belief that in almost no important cases would simpler rules be useful in reproducing the behavior of natural systems.

During the 1700s and 1800s there was ever-increasing success in using rules based on mathematical equations to analyze physical phenomena. And after the spectacular results achieved in physics in the early 1900s with mathematical equations there emerged an almost universal belief that absolutely every aspect of the natural world would in the end be explained by using such equations.

Needless to say, there were many phenomena that did not readily yield to this approach, but it was generally assumed that if only the necessary calculations could be done, then an explanation in terms of mathematical equations would eventually be found.

Beginning in the 1940s, the development of electronic computers greatly broadened the range of calculations that could be done. But disappointingly enough, most of the actual calculations that were tried yielded no fundamentally new insights. And as a result many people came to believe—and in some cases still believe today—that computers could never make a real contribution to issues of basic science.

But the crucial point that was missed is that computers are not just limited to working out consequences of mathematical equations. And indeed, what we have seen in this chapter is that there are fundamental discoveries that can be made if one just studies directly the behavior of even some of the very simplest computer programs.

In retrospect it is perhaps ironic that the idea of using simple programs as models for natural systems did not surface in the early days of computing. For systems like cellular automata would have been immensely easier to handle on early computers than mathematical equations were. But the issue was that computer time was an expensive commodity, and so it was not thought worth taking the risk of trying anything but well-established mathematical models.

By the end of the 1970s, however, the situation had changed, and large amounts of computer time were becoming readily available. And this is what allowed me in 1981 to begin my experiments on cellular automata.

There is, as I mentioned above, nothing in principle that requires one to use a computer to study cellular automata. But as a practical matter, it is difficult to imagine that anyone in modern times would have the patience to generate many pictures of cellular automata by hand. For it takes roughly an hour to make the picture on page 27 by hand, and it would take a few weeks to make the picture on page 29.

Yet even with early mainframe computers, the data for these pictures could have been generated in a matter of a few seconds and a few minutes respectively. But the point is that one would be very unlikely to discover the kinds of fundamental phenomena discussed in this chapter just by looking at one or two pictures. And indeed for me to do it certainly took carrying out quite large-scale computer experiments on a considerable number of different cellular automata.

If one already has a clear idea about the basic features of a particular phenomenon, then one can often get more details by doing fairly specific experiments. But in my experience the only way to find phenomena that one does not already know exist is to do very systematic and general experiments, and then to look at the results with as few preconceptions as possible. And while it takes only rather basic computer technology to make single pictures of cellular automata, it requires considerably more to do large-scale systematic experiments.

Indeed, many of my discoveries about cellular automata came as direct consequences of using progressively better computer technology.

As one example, I discovered the classification scheme for cellular automata with random initial conditions described at the beginning of Chapter 6 when I first looked at large numbers of different cellular automata together on high-resolution graphics displays. Similarly, I discovered the randomness of rule 30 (page 27) when I was in the process of setting up large simulations for an early parallel-processing computer. And in more recent years, I have discovered a vast range of new phenomena as a result of easily being able to set up large numbers of computer experiments in *Mathematica*.

Undoubtedly, therefore, one of the main reasons that the discoveries I describe in this chapter were not made before the 1980s is just that computer technology did not yet exist powerful enough to do the kinds of exploratory experiments that were needed.

But beyond the practicalities of carrying out such experiments, it was also necessary to have the idea that the experiments might be worth doing in the first place. And here again computer technology played a crucial role. For it was from practical experience in using computers that I developed much of the necessary intuition.

As a simple example, one might have imagined that systems like cellular automata, being made up of discrete cells, would never be able to reproduce realistic natural shapes. But knowing about computer displays it is clear that this is not the case. For a computer display, like a cellular automaton, consists of a regular array of discrete cells or pixels. Yet practical experience shows that such displays can produce quite realistic images, even with fairly small numbers of pixels.

And as a more significant example, one might have imagined that the simple structure of cellular automaton programs would make it straightforward to foresee their behavior. But from experience in practical computing one knows that it is usually very difficult to foresee what even a simple program will do. Indeed, that is exactly why bugs in programs are so common. For if one could just look at a program and immediately know what it would do, then it would be an easy matter to check that the program did not contain any bugs.

Notions like the difficulty of finding bugs have no obvious connection to traditional ideas in science. And perhaps as a result of this, even after computers had been in use for several decades, essentially none of this type of intuition from practical computing had found its way into basic science. But in 1981 it so happened that I had for some years been deeply involved in both practical computing and basic science, and I was therefore in an almost unique position to apply ideas derived from practical computing to basic science.

Yet despite this, my discoveries about cellular automata still involved a substantial element of luck. For as I mentioned on page 19, my very first experiments on cellular automata showed only very simple behavior, and it was only because doing further experiments was technically very easy for me that I persisted.

And even after I had seen the first signs of complexity in cellular automata, it was several more years before I discovered the full range of examples given in this chapter, and realized just how easily complexity could be generated in systems like cellular automata.

Part of the reason that this took so long is that it involved experiments with progressively more sophisticated computer technology. But the more important reason is that it required the development of new intuition. And at almost every stage, intuition from traditional science took me in the wrong direction. But I found that intuition from practical computing did better. And even though it was sometimes misleading, it was in the end fairly important in putting me on the right track.

Thus there are two quite different reasons why it would have been difficult for the results in this chapter to be discovered much before computer technology reached the point it did in the 1980s. First, the necessary computer experiments could not be done with sufficient ease that they were likely to be tried. And second, the kinds of intuition about computation that were needed could not readily have been developed without extensive exposure to practical computing.

But now that the results of this chapter are known, one can go back and see quite a number of times in the past when they came at least somewhat close to being discovered.

It turns out that two-dimensional versions of cellular automata were already considered in the early 1950s as possible idealized models for biological systems. But until my work in the 1980s the actual investigations of cellular automata that were done consisted mainly in constructions of rather complicated sets of rules that could be shown to lead to specific kinds of fairly simple behavior.

The question of whether complex behavior could occur in cellular automata was occasionally raised, but on the basis of intuition from engineering it was generally assumed that to get any substantial complexity, one would have to have very complicated underlying rules. And as a result, the idea of studying cellular automata with simple rules never surfaced, with the result that nothing like the experiments described in this chapter were ever done.

In other areas, however, systems that are effectively based on simple rules were quite often studied, and in fact complex behavior was sometimes seen. But without a framework to understand its significance, such behavior tended either to be ignored entirely or to be treated as some kind of curiosity of no particular fundamental significance.

Indeed, even very early in the history of traditional mathematics there were already signs of the basic phenomenon of complexity. One example known for well over two thousand years concerns the distribution of prime numbers (see page 132). The rules for generating primes are simple, yet their distribution seems in many respects random. But almost without exception mathematical work on primes has concentrated not on this randomness, but rather on proving the presence of various regularities in the distribution.

Another early sign of the phenomenon of complexity could have been seen in the digit sequence of a number like $\pi \simeq 3.141592653 \ldots$ (see page 136). By the 1700s more than a hundred digits of π had been computed, and they appeared quite random. But this fact was treated essentially as a curiosity, and the idea never appears to have arisen that there might be a general phenomenon whereby simple rules like those for computing π could produce complex results.

In the early 1900s various explicit examples were constructed in several areas of mathematics in which simple rules were repeatedly applied to numbers, sequences or geometrical patterns. And sometimes nested or fractal behavior was seen. And in a few cases substantially more complex behavior was also seen. But the very complexity of this behavior was usually taken to show that it could not be relevant for real mathematical work—and could only be of recreational interest.

When electronic computers began to be used in the 1940s, there were many more opportunities for the phenomenon of complexity to be seen. And indeed, looking back, significant complexity probably did occur in many scientific calculations. But these calculations were almost always based on traditional mathematical models, and since previous analyses of these models had not revealed complexity, it tended to be assumed that any complexity in the computer calculations was just a spurious consequence of the approximations used in them.

One class of systems where some types of complexity were noticed in the 1950s are so-called iterated maps. But as I will discuss on page 149, the traditional mathematics that was used to analyze such systems ended up concentrating only on certain specific features, and completely missed the main phenomenon discovered in this chapter.

It is often useful in practical computing to produce sequences of numbers that seem random. And starting in the 1940s, several simple procedures for generating such sequences were invented. But perhaps because these procedures always seemed quite ad hoc, no general conclusions about randomness and complexity were drawn from them.

Along similar lines, systems not unlike the cellular automata discussed in this chapter were studied in the late 1950s for generating random sequences to be used in cryptography. Almost all the results that were obtained are still military secrets, but I do not believe that any phenomena like the ones described in this chapter were discovered.

And in general, within the context of mainstream science, the standard intuition that had been developed made it very difficult for anyone to imagine that it would be worth studying the behavior of the very simple kinds of computer programs discussed in this chapter. But outside of mainstream science, some work along such lines was done. And for example in the 1960s early computer enthusiasts tried running various simple programs, and found that in certain cases these programs could succeed in producing nested patterns.

Then in the early 1970s, considerable recreational computing interest developed in a specific two-dimensional cellular automaton known as the Game of Life, whose behavior is in some respects similar to the rule 110 cellular automaton discussed in this chapter. Great effort was spent trying to find structures that would be sufficiently simple and predictable that they could be used as idealized components for engineering. And although complex behavior was seen it was generally treated as a nuisance, to be avoided whenever possible.

In a sense it is surprising that so much could be done on the Game of Life without the much simpler one-dimensional cellular automata in this chapter ever being investigated. And no doubt the lack of a connection to basic science was at least in part responsible.

But whatever the reasons, the fact remains that, despite many hints over the course of several centuries, the basic phenomenon that I have described in this chapter was never discovered before.

It is not uncommon in the history of science that once a general new phenomenon has been identified, one can see that there was already evidence of it much earlier. But the point is that without the framework that comes from knowing the general phenomenon, it is almost inevitable that such evidence will have been ignored.

It is also one of the ironies of progress in science that results which at one time were so unexpected that they were missed despite many hints eventually come to seem almost obvious. And having lived with the results of this chapter for nearly two decades, it is now difficult for me to imagine that things could possibly work in any other way. But the history that I have outlined in this section—like the history of many other scientific discoveries—provides a sobering reminder of just how easy it is to miss what will later seem obvious.